



Palm Computing' Platform

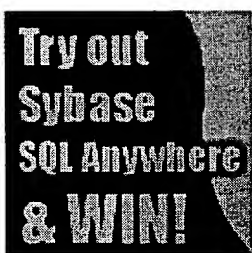
Development Zone

- ▷ [Developer Home](#)
- ▷ [Getting Started](#)
- ▷ [Documentation](#)
- ▷ [Knowledge Base](#)
- ▷ [Tools](#)
- ▷ [Developer Assistance](#)
- ▷ [Provider Pavilion](#)
- ▷ [www.palm.com](#)

Quick Index

Conduits
Creator IDs
Devsupp Email
Documentation
Email Support

Search



Creating .PQAs

Overview

This webpage covers creating query applications, .PQAs, for the Palm VII connected organizer. It is a distillation of the material covered in the 1998 Palm Developers Conference, and should not be confused with real documentation!

We strongly encourage you to [watch the presentation](#) online and use this page as written notes for it.

Outline

- [Technical Overview](#)
- [How to create a .PQA](#)
- [Creating Great .PQAs](#)

Technical Overview

Clipping Concepts

Web Clipping is a radically different concept from typical web browsing. Web Clipping is optimized for quick fact-lookup, not for researching or reading. Because the user is paying per byte, developers are strongly encouraged to give users the meat, not the fluff, of web content.

PQAs serve as applications, from the user's perspective. They are created with html, and are resident on the Palm VII device. They consist of some number of pages and optionally images: effectively a mini web site.

Here is a technical overview of how the process goes, from when the developer's starts with HTML through how the proxy works with a result web clipping.

Step 1

The developer creates html files, which will be the content resident on the device, in a .PQA. The Query Application Builder runs on the developer's desktop, and generates a Query Application (.PQA) file from that HTML. The Query Application Builder currently runs on Windows, though we plan to make it available for other platforms in the future. The source code for it

will be made available as well.

Step 2

The user installs a .PQA through the standard HotSync® process, and eventually will tap its icon in the Palm VII device launcher. This opens the .PQA in the Clipper application. (The user doesn't really know that the Clipper application exists; to them they're just opening a PQA. Clipper is the Palm application which displays the content of the PQA, much like a web browser does on a desktop machine.)

The user is now reading the information inside the PQA. This could be static data, could have multiple screens, hyperlinked together, and so forth. At some point they will usually fill out a form and taps on the submit button to wirelessly retrieve the data, or else will follow a link that takes them over the air. At this point, the Clipper application sends compressed packets out to the proxy servers. The proxy server contacts the "real" host via standard http or https. The host returns an html page, or web clipping, back to the proxy server. To the host, this is a normal html page access or cgi submission, just like would be received from a standard desktop browser. The mechanisms are standard; the content is just tuned to the needs of the Palm VII user.

Step 3

Now the proxy server will compress and tune the html result page (Web Clipping). The proxy will always remove unsupported HTML tags, frames, Java, animated GIFs, scripting code and the like from a page which is being sent to a Palm VII, since they aren't appropriate for the device. Whitespace, comments, and the actual text of HTML tags are not sent in the compressed web clipping, due to the custom compression performed on the page.

The result page (either a static html page or one generated by a cgi) can, with a meta tag, inform the proxy that it is a html page which is optimized for viewing on a Palm VII. If it hasn't, then the server will (in addition to the previously mentioned cleanup) remove all images and will truncate the web clipping to 1 K in order to safeguard the user from excessive airtime charges.

Parameter names in forms, however, are sent to the Palm VII device in full, so it is wise to use short names when possible, which will both reduce the user's cost as well as increase performance.

Step 4

The response is received by the Palm VII device and is displayed to the user. The web clipping may be added to the History list, with a title specified by the developer.

Keep in Mind

- Remember that the user is paying per byte, so be careful what you send back to the device! Usually you'll want to only send a subset of what is viewed by desktop browsers;

give them the data that is helpful and appropriate for their situation, and skip the fluff.

- Generally, developers will want to write their HTML by hand, since WYSIWIG editors are typically not very well suited to writing efficient HTML.
- https secure requests are supported, but the resulting packets are much larger (hence, more expensive), than normal http requests. It is absolutely fine to use them in appropriate circumstances, but do keep in mind the performance tradeoffs.

How to create a .PQA

Requirements

- Windows 95/98/NT
- Query Application Builder (supplied by Palm)
- A text editor
- HTML knowledge
- CGIs experience is also helpful, since .PQAs will often be using them, but it isn't required.

Supported Features

.PQAs support a subset of the HTML 3.2 specification; meaning most of the "normal" tags. This also includes:

- Tables (1 level deep)
- Fonts (Bold, Italic, and Larger font sizes)
- Form elements (Checkboxes, drop down lists, text fields, and radio buttons)

Features which are not supported in .PQAs include the following. Some of these might be implemented in the future, but others aren't appropriate for the user model and/or current hardware.

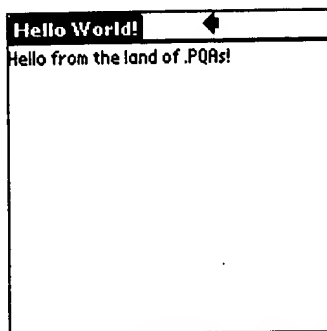
- Nested tables
- Frames
- Imagemaps
- Cookies
- JavaScript, Java, OCX, etc.

Sample .PQA: Hello World!

Here is an example of the HTML source used in a sample .PQA. Note that this is completely normal HTML.

```
<HTML>
<HEAD>
    <TITLE>Hello World!</TITLE>
</HEAD>
<BODY>
    Hello from the land of .PQAs!
</BODY>
</HTML>
```

Running that html through the Query Application Builder results in the following .PQA:



Special .PQA Elements

There are some special HTML elements that developers can use with their .PQAs. They include:

- Metatags
 - PalmComputingPlatform
 - HistoryListText
 - LocalIcon
 - PalmLauncherRevision
- Special form elements
 - Date and time pickers
- Special strings
 - %zipcode
 - %deviceid

Metatags

PalmComputingPlatform

Including this metatag indicates that the page is designed for a Palm VII device. Without it, images are removed and pages are truncated to 1024 bytes, to save the user from excessive airtime charges and unreadable pages. (See step 3, above.) The syntax for this metatag is:

```
<META NAME="PalmComputingPlatform" CONTENT="TRUE">
```

HistoryListText

This metatag is used with Web Clippings to store pages into the history list cache on a Palm VII device. The syntax for this metatag is:

```
<META NAME="HistoryListText" CONTENT="COMS Quote &date &time">
```

Developers may use the strings &date and &time to append the date and time to history list items.

LocalIcon

This metatag is used to store local images and pages in a .PQA. Web clippings can then reference these images and pages, obviating the need to transfer them over-the-air. The syntax for this metatag is:

```
<META NAME="LocalIcon" CONTENT="image.gif">
```

PalmLauncherRevision

This metatag is used to include a version number in a query application. Developers should include this metatag in the primary index of their .PQA. The syntax for this metatag is:

```
<META NAME="PalmLauncherRevision" CONTENT="X.X">
```

where X.X represents the version number.

Form Elements

Two form elements can be used to provide the most appropriate (and space-efficient) UI in your PQAs:

Date picker

Uses the native Palm OS's date picker to allow the user to enter a date.

```
<INPUT TYPE="DATEPICKER" NAME="date">>
```

- Returns Y2K compliant date: YYYY-MM-DD
- The current date is displayed as the default in the datepicker if VALUE is not provided

This is displayed on the form: **Date:**

Tapping it brings up this popup dialog:



Time picker

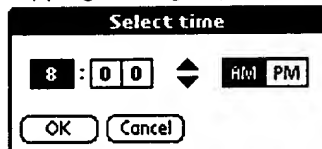
Uses the native Palm OS's time picker to allow the user to enter a time.

```
<INPUT TYPE="TIMEPICKER" NAME="date">>
```

- Returns time: hh:mm
- Current time is displayed in the timepicker if VALUE is undefined

This is displayed on the form: Time: 8:00 am

Tapping it brings up this popup dialog:



Special Strings

There are some special escape strings which can be added to a PQA to take advantage of some specific Palm VII features.

%zipcode

Used to provide information to the user based on her location at the time of the query. The value of this string is the zipcode of the nearest base station, which is usually within 5-10 miles from the user.

```
<INPUT TYPE="HIDDEN" NAME="zip" VALUE="%zipcode">>
```

%deviceid

Used to identify the person using the .PQA. The format of the return value is [-1,0,1].[x] where [-1,0,1] gives an indication of the server's certainty that the request is coming from a valid Palm VII device (1 means the server is certain, -1 means it knows it isn't coming over the air from a Palm VII device, 0 means the proxy isn't certain) and [x] is a unique string which identifies the device. The format of the data contained in the [x] portion should be considered "black-box" since it may change in the future.

```
<INPUT TYPE="HIDDEN" NAME="id" VALUE="%deviceid">>
```

- Example: 1.16300448.184812151

Referencing Pages and Images

When making a link within a pqa, meaning from local pages to local pages, you use relative referencing just the same way as you would between two html pages on the same server. For example:

- ``
- `` (In this case, the link is displayed as a button instead of an underlined link)
- ``

To link from a web clipping to a page or image which you know is in a PQA on the device, use the following URL syntax (substituting the name of your PQA for "app.pqa").

- ``
- ``

To link from a PQA (or web clipping) to a html page on a server, use the normal full URL syntax:

- `` (adds over-the-air icon to link)
- `` (adds secure over-the-air icon to link)

Creating Icons for Your .PQA

There are two icons to design for your .PQA. Both are displayed in the Applications launcher on your Palm VII device. Developers may save their icons in .bmp, .gif, or .jpg file formats.

Launcher Icon View icon

The large (32x22) icon should incorporate the diamond background, to indicate to the user that this is a PQA which could go over the air.



An example Launcher icon view:

Launcher List View icon

This icon is 15x9 pixels and should be similar to the larger icon, but does not need to contain the diamond due to the limited pixels available.

Here is a sample Launcher list view:

Creating Great .PQAs

The Zen of Palm

With designing .PQAs, less is definitely more. Developers should provide only the information that the user needs; remember that the user is paying for each byte that's sent over-the-air.

Developers should try to design their web clippings as small and clean as possible.

It's important to simplify the interface of the .PQA and web clippings. Developers should count the number of taps it takes to gather and retrieve information from their .PQA. Try placing infrequently used pages or complex features on sub-pages in a .PQA. It's not essential to implement everything on your website in your .PQA. Try to fit all the data on one screen, to prevent the user from having to scroll.

Use the LocalIcon metatag to store graphics and pages in your .PQA. Web clippings can simply reference these pages or images without having to send them over-the-air, a very cost prohibitive action.

[Development Zone Home](#) | [Getting Started](#) | [Documentation](#) | [FAQs](#) | [Knowledge Base](#) | [Tools](#) | [Developer Assistance](#) | [Provider Pavilion](#)

© 2000 Palm, Inc. All rights reserved. [Terms of Use](#).